Package 'pqrBayes'

December 21, 2024

Type Package

Title Bayesian Penalized Quantile Regression

Version 1.0.3

Date 2024-12-21

Description The quantile varying coefficient model is robust to data heterogeneity, outliers and heavy-tailed distributions in the response variable. In addition, it can flexibly model dynamic patterns of regression coefficients through nonparametric varying coefficient functions. In this package, we have implemented the Gibbs samplers of the penalized Bayesian quantile varying coefficient model with spike-and-slab priors [Zhou et al.(2023)]<doi:10.1016/j.csda.2023.107808> for efficient Bayesian shrinkage estimation, variable selection and statistical inference. In particular, valid Bayesian inferences on sparse quantile varying coefficient functions can be validated on finite samples. The Markov Chain Monte Carlo (MCMC) algorithms of the proposed and alternative models can be efficiently performed by using the package.

Depends R (>= 3.5.0)

License GPL-2

Encoding UTF-8

URL https://github.com/cenwu/pqrBayes

BugReports https://github.com/cenwu/pqrBayes/issues

LazyData true

Imports Rcpp,glmnet

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.3.2

NeedsCompilation yes

Repository CRAN

Author Cen Wu [aut, cre],
Kun Fan [aut],
Jie Ren [aut],
Fei Zhou [aut]

Maintainer Cen Wu <wucen@ksu.edu>

Date/Publication 2024-12-21 09:20:02 UTC

Contents

`pqrBayes-package` *Regularized Bayesian Quantile Varying Coefficient Model*

Description

In this package, we implement a sparse Bayesian quantile varying coefficient model for non-linear gene-environment interactions. The quantile varying coefficient functions that can capture the non-linear gene-environment interactions are approximated using B-splines. Quantile regression is adopted as it's robust to long-tailed distributions in the response/phenotype and provides the capability of describing the relationship between the response variable and predictors at different quantile leves. The default method, Bayesian regularized quantile varying coefficient model with spike-and-slab priors, adopts the point-mass spike–and–slab priors to achieve exact sparsity by shrinking the coefficients of unimportant effects to exactly zero and facilitate valid Bayesian inferences on quantile varying coefficients. In addition to the default method, users can also choose the method without robustness and spike–and–slab priors.

Details

The user friendly, integrated interface **pqrBayes()** allows users to flexibly choose the fitting methods by specifying the following parameter:

robust: whether to fit the robust sparse quantile varying coefficient models or the non-robust counterpart.

sparse: whether to use the spike-and-slab priors to impose sparsity.

The function pqrBayes() returns a pqrBayes object that contains the posterior estimates of each coefficients.

References

Zhou, F., Ren, J., Ma, S. and Wu, C. (2023). The Bayesian regularized quantile varying coefficient model. *Computational Statistics & Data Analysis*, 107808 doi:10.1016/j.csda.2023.107808

Ren, J., Zhou, F., Li, X., Ma, S., Jiang, Y. and Wu, C. (2023). Robust Bayesian variable selection for gene-environment interactions. *Biometrics*, 79(2), 684-694 doi:10.1111/biom.13670

Wu, C., and Ma, S. (2015). A selective review of robust variable selection with applications in bioinformatics. *Briefings in Bioinformatics*, 16(5), 873–883 doi:10.1093/bib/bbu046

Zhou, F., Ren, J., Lu, X., Ma, S. and Wu, C. (2021). Gene–Environment Interaction: a Variable Selection Perspective. *Epistasis. Methods in Molecular Biology.* 2212:191–223 https://link.springer.com/protocol/10.1007/978-1-0716-0947-7_13

Ren, J., Zhou, F., Li, X., Chen, Q., Zhang, H., Ma, S., Jiang, Y. and Wu, C. (2020) Semi-parametric Bayesian variable selection for gene-environment interactions. *Statistics in Medicine*, 39: 617– 638 doi:10.1002/sim.8434

Ren, J., Zhou, F., Li, X., Wu, C. and Jiang, Y. (2019) spinBayes: Semi-Parametric Gene-Environment Interaction via Bayesian Variable Selection. R package version 0.1.0. https://CRAN.R-project.org/package=spinBayes

Wu, C., Jiang, Y., Ren, J., Cui, Y. and Ma, S. (2018). Dissecting gene-environment interactions: A penalized robust approach accounting for hierarchical structures. *Statistics in Medicine*, 37:437–456 doi:10.1002/sim.7518

Wu, C., Shi, X., Cui, Y. and Ma, S. (2015). A penalized robust semiparametric approach for gene-environment interactions. *Statistics in Medicine*, 34 (30): 4016–4030 doi:10.1002/sim.6609

Wu, C., Cui, Y., and Ma, S. (2014). Integrative analysis of gene–environment interactions under a multi–response partially linear varying coefficient model. *Statistics in Medicine*, 33(28), 4988–4998 doi:10.1002/sim.6287

Wu, C., Zhong, P.S. and Cui, Y. (2018). Additive varying–coefficient model for nonlinear gene–environment interactions. *Statistical Applications in Genetics and Molecular Biology*, 17(2) doi:10.1515/sagmb-20170008

Wu, C., Zhong, P.S. and Cui, Y. (2013). High dimensional variable selection for gene-environment interactions. *Technical Report. Michigan State University.*

See Also

pqrBayes

`data` *simulated data for demonstrating the features of pqrBayes*

Description

Simulated gene expression data for demonstrating the features of pqrBayes.

Format

The data object consists of five components: g, y, u, e and coeff. coeff contains the true values of parameters used for generating the response variable y.

Details

The model for generating Y

Use subscript i to denote the ith subject. Let $(\boldsymbol{X}_i, Y_i, V_i, \boldsymbol{E}_i)$, $(i = 1, \ldots, n)$ be independent and identically distributed random vectors. Y_i is a continuous response variable representing the disease phenotype. $\boldsymbol{X}_i = (X_{i0}, ..., X_{ip})^\top$ denotes a $(1+p)$–dimensional vector of predictors (e.g. genetic factors) with the first element $X_{i0} = 1$. The environmental factor $V_i \in \mathbb{R}^1$ is a univariate index variable. $\boldsymbol{E}_i = (E_{i1}, ..., E_{iq})^\top$ is the q-dimensional vector of clinical covariates. At a given quantile level τ, considering the following quantile varying coefficient model:

$$Y_i = \sum_{k=1}^{q} E_{ik}\beta_{k,\tau} + \sum_{j=0}^{p} \gamma_{j,\tau}(V_i)X_{ij} + \epsilon_{i,\tau},$$

where $\beta_{k,\tau}$'s are the regression coefficients for the clinical covariates and $\gamma_{j,\tau}(\cdot)$'s are unknown smooth varying-coefficient functions. The regression coefficients of $\boldsymbol{X}$ vary with the univariate index variable $\boldsymbol{v} = (v_1, ..., v_n)^\top$. The $\epsilon_{i,\tau}$ is the random error. For simplicity of notation, the quantile level τ has been suppressed hereafter.

The true model that we used to generate Y:

$$Y_i = \gamma_0(v_i) + \gamma_1(v_i)X_{i1} + \gamma_2(v_i)X_{i2} + \gamma_3(v_i)X_{i3} + \epsilon_i,$$

where $\epsilon_i \sim N(0,1)$, $\gamma_0 = 1.5\sin(0.2\pi * v_i)$, $\gamma_1 = 2\exp(0.2v_i - 1) - 1.5$, $\gamma_2 = 2 - 2v_i$ and $\gamma_3 = -4 + (v_i - 2)^3/6$.

See Also

pqrBayes

Examples

```
data(data)
g=data$g
dim(g)
coeff=data$coeff
print(coeff)
```

---

pqrBayes *fit a regularized Bayesian quantile varying coefficient model*

Description

fit a regularized Bayesian quantile varying coefficient model

Usage

```
pqrBayes(
  g,
  y,
  u,
  e = NULL,
  quant = 0.5,
  iterations = 10000,
  kn = 2,
  degree = 2,
  robust = TRUE,
  sparse = TRUE,
  hyper = NULL,
  debugging = FALSE
)
```

Arguments

g	the matrix of predictors (subject to selection) without intercept.
y	the response variable. The current version only supports the continuous response.
u	a vector of effect modifying variable of the quantile varying coefficient model.
e	a matrix of clinical covariates not subject to selection.
quant	the quantile level specified by users. The default value is 0.5.
iterations	the number of MCMC iterations.
kn	the number of interior knots for B-spline.
degree	the degree of B-spline basis.
robust	logical flag. If TRUE, robust methods will be used.
sparse	logical flag. If TRUE, spike-and-slab priors will be used to shrink coefficients of irrelevant covariates to zero exactly.
hyper	a named list of hyperparameters.
debugging	logical flag. If TRUE, progress will be output to the console and extra information will be returned.

Details

The model described in "data" is:

$$Y_i = \sum_{k=1}^{q} E_{ik}\beta_k + \sum_{j=0}^{p} \gamma_j(V_i)X_{ij} + \epsilon_i,$$

where β_k's are the regression coefficients for the clinical covariates and γ_j's are the varying coefficients for the intercept and predictors (e.g. genetic factors).

When {sparse=TRUE} (default), spike–and–slab priors are adopted. Otherwise, Laplacian shrinkage will be used.

When {robust=TRUE} (default), quantile varying coefficient models are adopted. Otherwise, the least-square based varying coefficient model will be used.

Users can modify the hyper-parameters by providing a named list of hyper-parameters via the argument 'hyper'. The list can have the following named components

a0, b0: shape parameters of the Beta priors ($\pi^{a_0-1}(1-\pi)^{b_0-1}$) on π_0.

c1, c2: the shape parameter and the rate parameter of the Gamma prior on ν.

Please check the references for more details about the prior distributions.

Value

an object of class "pqrBayes" is returned, which is a list with components:

`posterior`	posterior samples from the MCMC
`coefficients`	a list of posterior estimates of coefficients

References

Zhou, F., Ren, J., Ma, S. and Wu, C. (2023). The Bayesian regularized quantile varying coefficient model. *Computational Statistics & Data Analysis*, 107808 doi:10.1016/j.csda.2023.107808

Ren, J., Zhou, F., Li, X., Ma, S., Jiang, Y. and Wu, C. (2023). Robust Bayesian variable selection for gene-environment interactions. *Biometrics*, 79(2), 684-694 doi:10.1111/biom.13670

Ren, J., Zhou, F., Li, X., Chen, Q., Zhang, H., Ma, S., Jiang, Y. and Wu, C. (2020) Semi-parametric Bayesian variable selection for gene-environment interactions. *Statistics in Medicine*, 39: 617– 638 doi:10.1002/sim.8434

Examples

```
data(data)
g=data$g
y=data$y
u=data$u
e=data$e

## default method
fit1=pqrBayes(g,y,u,e,quant=0.5)
fit1



## non-sparse
sparse=FALSE
fit2=pqrBayes(g,y,u,e,quant=0.5,sparse = sparse)
fit2

## non-robust
robust = FALSE
fit3=pqrBayes(g,y,u,e,quant=0.5,robust = robust)
fit3
```

predict.pqrBayes *make predictions from a pqrBayes object*

Description

make predictions from a pqrBayes object

Usage

```
## S3 method for class 'pqrBayes'
predict(
  object,
  g.new,
  u.new,
  e.new = NULL,
  y.new = NULL,
  quant = 0.5,
  kn = 2,
  degree = 2,
  ...
)
```

Arguments

object	pqrBayes object.
g.new	a matrix of new predictors (e.g. genetic factors) at which predictions are to be made.
u.new	a vector of new environmental factor at which predictions are to be made.
e.new	a vector or matrix of new clinic covariates at which predictions are to be made.
y.new	a vector of the response of new observations. If provided, the prediction error will be computed based on Y.new.
quant	the quantile for the response variable. The default is 0.5.
kn	the number of interior knots for B-spline.
degree	the degree of B-spline basis.
...	other predict arguments

Details

g.new (u.new) must have the same number of columns as g (u) used for fitting the model. By default, the clinic covariates are NULL unless provided. The predictions are made based on the posterior estimates of coefficients in the pqrBayes object.

If y.new is provided, the prediction error will be computed based on the check loss.

Value

an object of class 'pqrBayes.pred' is returned, which is a list with components:

`error`	prediction error. error is NULL if y.new=NULL.
`y.pred`	predicted values of the new observations.

See Also

`pqrBayes`

`print.pqrBayes` *print a pqrBayes result*

Description

Print a pqrBayes result

Usage

```
## S3 method for class 'pqrBayes'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`x`	pqrBayes result
`digits`	significant digits in printout.
`...`	other print arguments

Value

No return value, called for side effects.

See Also

`pqrBayes`

print.pqrBayes.pred *print a pqrBayes.pred object*

Description

Print a summary of a pqrBayes.pred object

Usage

```
## S3 method for class 'pqrBayes.pred'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

x	pqrBayes.pred object.
digits	significant digits in printout.
...	other print arguments

Value

No return value, called for side effects.

See Also

predict.pqrBayes

print.VCselect *print a select.VC object*

Description

Print a summary of a select.VC object

Usage

```
## S3 method for class 'VCselect'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

x	VCselect object.
digits	significant digits in printout.
...	other print arguments

Value

No return value, called for side effects.

See Also

VCselect

`VCselect` *Variable selection for a pqrBayes object*

Description

Variable selection for a pqrBayes object

Usage

```
VCselect(obj, sparse, iterations = 10000, kn = 2, degree = 2)
```

Arguments

`obj`	pqrBayes object.
`sparse`	logical flag.
`iterations`	the number of MCMC iterations.
`kn`	the number of interior knots for B-spline.
`degree`	the degree of B-spline basis.

Details

For class 'Sparse', the median probability model (MPM) (Barbieri and Berger, 2004) is used to identify predictors that are significantly associated with the response variable. For class 'NonSparse', variable selection is based on 95% credible interval. Please check the references for more details about the variable selection.

Value

an object of class 'VCselect' is returned, which includes the indices of the selected predictors (e.g. genetic factors).

References

Ren, J., Zhou, F., Li, X., Ma, S., Jiang, Y. and Wu, C. (2023). Robust Bayesian variable selection for gene-environment interactions. *Biometrics*, 79(2), 684-694 doi:10.1111/biom.13670

Barbieri, M.M. and Berger, J.O. (2004). Optimal predictive model selection. *Ann. Statist*, 32(3):870–897

See Also

pqrBayes

Examples

```
data(data)
g=data$g
y=data$y
u=data$u
e=data$e
## default method
fit1=pqrBayes(g,y,u,e,quant=0.5)
sparse=TRUE
select=VCselect(obj = fit1,sparse = sparse)
select


## non-sparse
sparse=FALSE
fit2=pqrBayes(g,y,u,e,quant=0.5,sparse = sparse)
select=VCselect(obj=fit2,sparse=FALSE)
select
```

Index